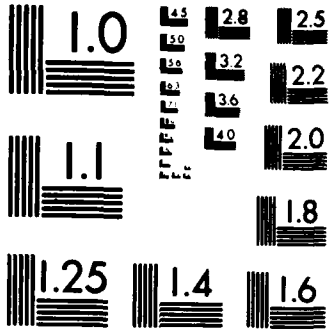
AD-A190 270 AUTOMATED MEDICAL ACQUISITION IN FIELD MEDICAL SYSTEMS 1/1
(U) NAVAL HEALTH RESEARCH CENTER SAN DIEGO CA
T J SAGER ET AL 10 DEC 87 NHRC-87-35

UNCLASSIFIED F/G 12/7 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

Naval Health Research Center

DTIC FILE COPY

AD-A190 270

AUTOMATED MEDICAL ACQUISITION IN FIELD MEDICAL SYSTEMS

T. J. SAGER
W. M. PUGH

DTIC
ELECTE
FEB 05 1988
S D
H

REPORT NO. 87-35

Approved for public release: distribution unlimited.

NAVAL HEALTH RESEARCH CENTER
P.O. BOX 85122
SAN DIEGO, CALIFORNIA 92138

NAVAL MEDICAL RESEARCH AND DEVELOPMENT COMMAND
BETHESDA, MARYLAND

Automated Medical Acquisition in Field Medical Systems*

Dr. Thomas J. Sager
Department of Computer Science
Mathematical Sciences Building
University of Missouri-Columbia
Columbia, MO 65211

and

William M. Pugh
Medical Information Systems Department
Naval Health Research Center
P. O. Box 85122
San Diego, CA 92138-9174

Accession For	
NTIS GRA&I	☑
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

*Report Number 87-35, supported by the Naval Medical Research and Development Command, Bethesda, Maryland, Department of the Navy under research work unit 63706N M0095.005-1053. The opinions expressed in this article are those of the author and do not reflect the official policy or position of the Department of the Navy, Department of Defense, nor the U.S. Government. Approved for public release, distribution unlimited.

This research was a collaborative effort between the Naval Health Research Center and the University of Missouri under the Navy/American Society for Engineering Education Summer Faculty Research Program.

88 2 01 045

Summary

For field operational settings, automated systems are being evaluated to process medical information. In such environments, the efficient acquisition and storage of information is highly important. Therefore, a prototype system using a graphics tablet interfaced with a microcomputer was designed and developed to explore the utility of automated techniques for capturing medical data in field environments.

The functional design specified that the system should allow new input forms to be entered into the system by using the graphics tablet to define areas which were then associated with semantic information defined on the form. It was also specified that the user should be able to modify both the definitions and associated areas; that a review capability would be provided for form designers; that inconsistent area definitions would be identified by the system; that after review, forms could be filed; and that the system would be menu driven.

A prototype system was developed in Turbo Pascal on an IBM PC/AT compatible computer. The software allowed x,y coordinate information from the graphics tablet, which defined rectangular areas, to be associated with user definitions of those areas. To implement this capability in a manner that would execute efficiently, quadcodes were used. The quaternary number 1 represented the entire tablet, and the numbers 10, 11, 12, and 13 were used to represent the upper left quadrant, upper right quadrant, lower left quadrant, and lower right quadrant, respectively. Further, each quadrant could be broken down recursively into four subquadrants.

The prototype system demonstrated that a graphic data capture system is an effective method for capturing medical information. Continued testing is recommended to explore the full potential of this capability.

Automated Medical Acquisition in Field Medical Systems

Dr. Thomas J. Sager and William M. Pugh

The conceptual design and initial testing of a casualty care information system[1,5] for the U.S. Marine Corps has been conducted by the Naval Health Research Center (NHRC). Although the activity of medical personnel is primarily directed to patient care in combat situations, having up-to-date records can make the difference between life and death. Therefore, one goal of medical systems developed for field environments is the acquisition of data in a manner that does not detrimentally impact the ability of medical personnel to provide emergency treatment at the forward echelons.

One of the methods of capturing data being considered is the use of a graphics tablet attached to a microcomputer. In this method, standard forms will be designed with standardized information printed on them. The examining physician or his assistant will check off information on the form, writing in comments only where applicable and where time and work load permit. Information from these forms will then be entered into the computer system by corpsmen. Before moving a patient to another facility further from combat, his medical data will be downloaded from the microcomputer onto some machine readable medium which will then travel with the patient.

To enter data from a standardized form, the corpsman positions the form on the graphics tablet and touches the area of the form checked by the physician with a wand or mouse. The graphics tablet merely sends the (x,y) coordinates of the position on the tablet touched by the wand to the microcomputer. Software within the computer must translate the (x,y) coordinates into the proper semantic information as defined on the form. Only the physician's comments and certain alpha-numeric data need be entered from the keyboard.

It is foreseen that standardized forms will change with time and advances in the medical field. It is also anticipated that the problem of designing a form for use under battlefield conditions will not be a simple process. The form designers will probably have to make multiple passes before generating a form that contains all the required information laid out in a manner that is convenient to both the physician and the entry person.

For these reasons, we felt it would be very helpful, if not absolutely necessary, to have a Computer-Aided Design (CAD) system that would assist the designer in creating both standardized forms and the tables of data and subprograms necessary to enter data from these forms into a computer system via a graphics tablet.

Below is a description of the design. The prototype implementation was done on an IBM PC/AT using a Summagraphics MM 1201 graphics tablet. The software, including a driver and interrupt handler to manage communications between the PC and the graphics tablet, was written entirely in Borland International's Turbo Pascal. The prototype implementation contains most of the features described in the design and is currently under evaluation.

Design

On a functional level, the following system attributes were identified:

1. The primary purpose of the system is to allow the designer to define certain areas of a form to have different meanings.

2. The system should allow the designer to easily change the meanings of areas as well as to create, delete, move, expand or contract areas.

3. The system should not only keep track of areas already defined but should also keep the form designer aware of how the layout is progressing.

4. The system should be able to detect and notify the designer of inconsistencies such as overlapping areas or areas that are too small or too close to other areas.

5. On completion of the design of a form, the system should create a file of tables and subprograms which will allow a driver to translate a pair of coordinates on the graphics tablet to a given meaning. This file should be a source language file that could be manipulated with an editor included within a driver, or in a compiled form, called as an external program.

6. To ease the work of the form designer, the design process should be menu driven and user-friendly.

In addressing the problem of assigning meanings to areas, it was decided that for ease of internal representation, all defined areas should be rectangles aligned on the x and y axes. Non-rectangular areas, however, could be simulated as the union of more than one rectangular area. Also, it was considered that it might be convenient to define multiple areas on different parts of a form (or even areas on separate forms) to have the same meaning. Thus, it was decided that each rectangle would be assigned a semantic code which would signify the meaning of the area. Many different areas may be assigned the same code.

In addition, we allowed for giving many codes an aggregate name. Thus, a code may belong to a set. For example, KNEE, EYE and NECK may all be defined to belong to the set BODYPARTS; BURN, FRACTURE, and LACERATION may all be defined to belong to the set INJURY. This would allow, for example, a driver program to ask for the entry of (BODYPART, INJURY) pairs or (BODYPART, INJURY, TREATMENT) triples and easily verify that exactly one member of each set was entered before allowing the entry person to continue.

The problem of internal representation of rectangles was also addressed in the design. To facilitate the detection of overlapping areas or areas which are too small or narrow for convenient entry, it was decided to store each rectangle as a five-tuple: the x,y coordinates of the center, the distance from the center to the edge in both the x and y directions, and a code signifying the meaning of the area.

The problem of notifying the designer of exceptional conditions was a little more difficult. The design calls for a video image of the graphics tablet with the form fixed on top of it to be displayed upon a medium resolution color monitor in black and white. As areas of the form are defined, these areas are outlined in color on the monitor with the code written inside the area. Problem areas such as overlaps and exceptionally small or narrow areas would be highlighted in a different color, thus bringing them to the

designer's attention. The designer would only need to look at the monitor to be aware of the progress of the design.

For modification and deletion of areas, the designer should be able to select an area through the use of a mouse by moving the crosshairs to within the area of the monitor. Once an area is selected, functions will be provided for moving the area in any direction, shrinking or expanding it in either the x or y direction, deleting it, or changing its semantic code.

To create an area, the designer need only move the crosshairs of the mouse to the approximate location of the area and invoke the create function by pressing the appropriate key. The new area can then be manipulated through the modification functions. Alternately, the designer can create an area by tracing the boundary. Using this method, the area is transformed into a union of rectangles based on a tolerance parameter. The area traced is decomposed into as few rectangles as possible subject to the constraint that the actual boundary be always within the tolerance parameter of the boundary as traced.

The problem of translation from an (x,y) coordinate pair to a semantic code belongs as much to the subprogram created by the CAD system to translate (x,y) coordinates on the tablet to meaningful codes as it does to the CAD system itself. The translation subprogram must be reasonably fast and cannot require an impossibly large amount of memory.

The design calls for implementing the translation with quadcodes. Quadcodes and quadtrees have been discussed extensively in the literature. An excellent survey of the literature on quadtrees can be found in Samet[4]. A quadcode is a base four number. In our implementation, the quaternary number 1 stands for the entire tablet. The quaternary numbers 10, 11, 12 and 13 stand for the upper left quadrant, upper right quadrant, lower left quadrant and lower right quadrant respectively. Each quadrant may then be broken down recursively into four subquadrants. For example, 112 stands for the lower left subquadrant of the upper right quadrant. The translation from (x,y) coordinates to quadcodes is reasonably straight forward and is described by Li[2,3].

The design calls for partitioning the tablet space into as few quadcodes as possible, subject to the constraint that no quadcode intersects with more than a threshold number (perhaps 5) of defined rectangles. Once a point entered from the tablet is translated to a quadcode, the few candidate rectangles intersecting that quadcode are then searched linearly until the correct area, and thus the semantic code signifying the meaning of the area, is found.

Another problem that had to be addressed was one of alignment of a form on the graphics tablet and minor variations from copy-to-copy of a form due to the printing process. It was desired that the system should function properly regardless of minor variations in printing and alignment. To accomplish this, it was decided that all forms would be designed with two points printed on them: an origin in the lower left-hand corner and a reference point in the upper right-hand corner.

Before entering information from a form, the entry person or designer will be required to touch the origin and the reference point with a wand or mouse. From the absolute location of the origin and the reference point, and the expected distance in the x and y direction of the reference point from the origin, it is possible to transform an absolute location on the tablet to the position it would hold if the form had been aligned correctly with the origin at (0,0) and reference point at its expected location. If the reference point is not within a reasonable tolerance of its expected distance from the origin, the designer or entry person is asked to reposition the form and reenter the origin and reference point again before proceeding. If the reference point is within this expected tolerance, then all points entered from the tablet are translated into their expected offset in the x and y directions from the origin before being processed further.

Once the design of a form is completed, the system will create a source language (TBL) file containing the data and procedures necessary for efficient translation from an absolute point on the graphics tablet to the semantic information associated with the area in which the point lies. This TBL file supplies three procedures, DEFINE_ORIGIN, CHECK_REFPOINT and LOOKUP_POINT, which are described in Figure 1.

```
type point = record x,y   0..maxint end;

    { absolute coordinates from graphics tablet          }

procedure  DEFINE_ORIGIN  (p: point);

    { assigns the variable origin:  point; the value p   }

procedure  CHECK_REFPOINT (p: point);

    { checks  whether the distance between the points,  p }
    { and  origin, is within an acceptable tolerance  and }
    { computes  the parameters necessary to translate  an }
    { absolute  point to  a point relative to the  origin }
    { of the form when correctly positioned.             }

procedure LOOKUP_POINT    (p: point);

    { The parameter, number, is assigned -1 if the point, }
    { p, does not lie in a defined rectangle.  Otherwise }
    { number is assigned  a  non-negative  internal  code }
    { uniquely identifying  the  semantic code associated }
    { with the rectangle in which p lies. This number can }
    { be  used  in  the calling  routine  to retrieve the }
    { semantic code  and  other  associated semantic data }
    { from a table  or to check whether this code belongs }
    { to a specific set.  The table of semantic data and }
    { the set declarations are also placed in this file   }
```

Figure 1. Procedures on TBL file

Implementation

To test the feasibility of the concept design, a rapid development strategy was used to generate a prototype system. Therefore, this initial development used only a microcomputer and graphics tablet. As a result design features requiring a video camera and large medium resolution color monitor were not developed. The prototype system was implemented in five layers, each layer having its own menu. The outermost layer is the COMMAND layer. Progressing inward, the layers are called ENTRY, PARAMETER, REGION and SET.

Figure 2 shows the menu for the command layer. By entering a single character the designer may move inward into any of the other four menus. The characters "Q" and "H" have the standard meaning of Quit and Help throughout the system.

Entering the character "A" causes the regions defined so far to be analyzed for overlap or close proximity. Depending on a parameter which may be entered from the PARAMETER menu, overlapping regions may be shrunk to eliminate overlap. Tables to implement the LOOKUP_POINT procedure described above are then created. The character "T" allows the designer to test his design. He may now enter points from the graphics tablet, and the semantic data associated with the area containing the point entered is displayed on the screen. The character "M" causes a source code file with the extension TBL containing the data and procedures described above to be created. The characters "F" and "W" cause the regions and sets defined so far to be fetched from or written to a file with extension, SRC. Thus, the design definition can be saved and retrieved at will by the designer.

The other four layers have similar menus. The ENTRY menu is for defining regions; the PARAMETER menu is for viewing or modifying system parameters; and the REGION and SET menus are for viewing and modifying the regions or sets already defined.

COMMAND MENU

MENUS	TABLES
E - enter regions	A - analyze and create tables
P - view or modify parameters	T - test tables
R - view or modify regions	M - make .TBL file
SOURCE (.SRC) FILES	**OTHER COMMANDS**
F - fetch data	Q - quit
W - write data	H - help

Filename: **TEST**

ENTER SINGLE CHARACTER, ANY CASE_

Figure 2. Command Menu

Conclusions

A graphic data capture system can be used as an effective and efficient method for the capture of medical information in field settings. Continued testing of the prototype implementation and its application in the field tests should be pursued to explore the full potential of this capability.

REFERENCES

1. Congleton, M., Wilcox, W.W., Pugh, W.M., and Bartok, A. Information flows at the forward echelons of casualty care (NHRC Report 86-23). Naval Health Research Center, San Diego, CA, 1986.

2. Li, S.X. and Loew, M.H. The quadcode and its arithmetic. Commun. ACM 30, 7, July 1987, 621-626.

3. Li, S.X. and Loew, M.H. The quadcode and its arithmetic. Commun. ACM 30, 7, July 1987, 627-631.

4. Samet, H. The quadtree and related hierarchical data structures. ACM Comput. Surv. 16, 2 (June 1984) 187-260.

5. Wilcox, W.W. and Pugh, W.M. Combat casualty care medical information system (CCC/MIS) protoype (NHRC Report 86-32). Naval Health Research Center, San Diego, CA 1986.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

A190 270

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION Unclassified	1b RESTRICTIVE MARKINGS None
2a SECURITY CLASSIFICATION AUTHORITY N/A	3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited
2b DECLASSIFICATION/DOWNGRADING SCHEDULE N/A	
4 PERFORMING ORGANIZATION REPORT NUMBER(S) NHRC Report No1 87-35	5 MONITORING ORGANIZATION REPORT NUMBER(S)
6a NAME OF PERFORMING ORGANIZATION Naval Health Research Center 6b OFFICE SYMBOL (If applicable) Code 20	7a NAME OF MONITORING ORGANIZATION Commander, Naval Medical Command
6c. ADDRESS (City, State, and ZIP Code) P. O. Box 85122 San Diego, CA 92138-9174	7b. ADDRESS (City, State, and ZIP Code) Department of the Navy Washington, DC 20372
8a NAME OF FUNDING/SPONSORING ORGANIZATION Naval Medical Research & Development Command 8b OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code) Naval Medical Command National Capitol Region Bethesda, MD 20814-5044	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO: 63706N PROJECT NO: M0095 TASK NO: .05 WORK UNIT ACCESSION NO: DN246555

11 TITLE (Include Security Classification)

(U) AUTOMATED MEDICAL ACQUISITION IN FIELD MEDICAL SYSTEMS

12 PERSONAL AUTHOR(S)
Sager, T.G., Pugh, W.M.

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Final	FROM _____ TO _____	87 DEC 10	

16 SUPPLEMENTARY NOTATION This research was a collaborative effort between the Naval Health Research Center and the University of Missouri under the Navy/American Society for Engineering Education Summer Faculty Research Programs.

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	casualty care; automated input; computer-aided design

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

The functional design of a system to automatically capture combat casualty care data was defined. A prototype system was developed using a graphics tablet interfaced with an IBM PC/AT compatible computer. Software written in Turbo Pascal allowed users to associate areas selected from a form with appropriate semantic information. This capability was implemented using quadcodes which provided the degree of efficiency needed for timely data processing.

20 DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION Unclassified
22a NAME OF RESPONSIBLE INDIVIDUAL William M. Pugh	22b TELEPHONE (Include Area Code) 619/553-8403 22c OFFICE SYMBOL Code 20

DD FORM 1473, 84 MAR

83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

☆U.S. Government Printing Office: 1985—867-047

END
DATE
FILM
4-88
DTIC